Software Processes and Management
Assignment 1
Ao Li 798657

1. Introduction

In this essay, it agrees on using a prototype-driven, incremental software development lifecycle model on the Aura ----- Virtual Temporal Bone Surgery Project, cited in both $Team\ Aura, 2011, SRS^{[1]}$ and $Team\ Aura, 2011, SPMP^{[2]}$. In the following few parts will introduce the reasons for choosing incremental lifecycle model, features and requirements make the project difficult, potential risks during the plan, discuss why not choose scrum model and conclusion.

2. Reasons for Choosing Incremental Lifecycle Model

There are many reasons for choosing prototype-driven incremental lifecycle model to Virtual Surgery Project. Some of the important reasons shown as below:

1) The client has given some requirements that are classified into core requirements and non-core requirements, so the system is easy to divide them into different increments (Separated into two increments in the SRS).
2) Compare the non-core requirements with core requirements, non-core requirements aren't something new but extension function for the core requirements, meaning there are not exactly "novel features" added into the system. So it indicates that all clients' requirements are based on their core requirements. And in incremental model, generally, it needs to do all the requirements and most probably the architecture upfront, so the incremental model is satisfied with the project.
3) Clients can get some feedbacks from the core requirements increment. In other words, customer can see some features before the end of the product. So if something needs to be improved, the model allows clients add it into the system in the

later increment. For example, to “3D Virtual Surgery Screen” requirement, there are not values control with the zoom control box and it is implemented with the magnification range between 0~200%”. But maybe when magnify 200%, the limitation sharpness can’t meet clients’ requirement and the range need to be changed.

4) Since the non-core requirements is extension for the core requirements. To some extent, it indicates some of the resources can be better utilized when using incremental model. For example, some analysis data for the system in the first increment can be used into the second one.

5) Besides, due to repeatedly using resource and the second increment can be started before the first one completely finished. They can let the project be completed in a shorter time line. Because the Deliver Plan for the product is scheduled on October 30th, 2001, but the SRS document was written in April 7th, 2001, there are not much time for implementing the proposed system and model has a shorter time line is more appropriate.

3. Important Features and Requirements

Examples for some features and requirements make the project difficult are as below:

To Feature:

1) Temple bone surgery is such a difficult surgery and is a continuous process. It would be ideal that no errors occur in the system that would interrupt the simulated operation, but it’s hard to maintain the system’s reliability all the time.

2) As mention in SPMP, this project is being undertaken as a university subject and no formal budget is allocated. The limitation of the budget leads to many problems (e.g. The employees’ and techniques’ quality).

To Requirement:

1) The client has not requirement on indicating a preference for a particular programming language. It is more difficult for developer sometimes when there is not a specified language, since people need to think which language they can handle and judge which one is better for this particular project this kind of questions.
2) In order to refine requirement, need somebody have regular meeting with clients. But clients may not always be available or helpful.
3) The requirement "Zoom a 3D model" is such a difficult requirement for the system, which need to use a relatively advanced technique for zooming, otherwise the 3D model will not quite true to the original.

4. Risks during the Plan

There are some potential risks that contains in the project and some of them are shown as follow:

1) Technique Knowledge Risk:

Assuming people in the development group are inexperience, it would be hard to manage them. For example, they may have a limited knowledge of the programming language that will be used to implement the system.

2) Professional Knowledge Risk:

It's a system for temple bone surgery, such a small and specific medical domain, so most of the people in the whole team are not familiar with this professional area, The system is more likely to have some flaws that clients would highly be not satisfied with the final system.

3) Schedule Management Risk:

To incremental model, the second increment can be started before the first one is finished. So each team member may be part of more than one subteam and development group at a time.

Besides, the user documentation for the first increment may be produced during increment two, so maybe it's too late for clients to refine core requirements in the second increment after clients read the documents. It's a risk to make a schedule for scheduling time and people for each phase in each increment.

5. Reasons for not choosing Scrum Model

The most competitive lifecycle model for this project is Agile, take scrum as example. There are some advantages for using scrum:

1) **Experience Reference:** Since there are not big difference between non-core requirements and core requirements, so the sprint retrospective meeting in agile model is very helpful in this way, the team can discuss what improvement they can make for future sprints. But there is not this kind of meeting in incremental model for leaning experience.
2) **Easy for changeable environment:** Also, In Scrum, the clients' requirements can be separated into multiple product backlogs, since the product backlog can be changed at any time, so the model is much readier to react to the requirements modified by clients than prototype incremental model.
3) **Resource sharing:** Because there are much more meetings in scrum model than incremental model, it seems that they can share more things in scrum model (e.g. experience and technique support). Besides, there are many methods for working in a sprint, such as pair programming, which can help improving the working efficiency.

Agile has many good characteristics, but it's still not suitable enough for some of the important features of this project,

1) **Not add novel requirement:** From SRS, it has a clear requirements from clients and the system is just for bone surgery, so it can be regarded as it won't have novel requirements added into the project so that not need to use agile.

2) **Team size:** There are 17 people in team Aura, but to the number of people in a scrum team, generally it's about 3-9 people. So, apparently, team Aura need to be divided into two scrum teams. Multiple teams will occur many problems, such as communication problem, they may need to contact with clients more frequently than one team for different negotiation how the two teams communicate with each other is still a problem.

3) **Self-organized:** In scrum, it gives more responsibility to team members, which is called self-organized. But it's not a good idea for an inexperience team, because it will take a while before the process becomes effective.

6. Conclusion

Overall, prototype incremental lifecycle model is more suitable for this project than others. However, there are still some small problems mentioned in the documents, which is not a problem for the model itself, but a problem for applying the project to the model.

1) **The number of increment:**

The number of increment is an important part in incremental model. In SPMP, it just separated the original project into two increments, which is a little unreasonable. Since each increment contains too many requirements that makes the workload for each increment is so heavy and would get a higher risks for unsuccessful processes. Besides, it just contains two increments so that the number of iteration process is very few in this incremental model in order to make the project has few testing phase and clients can only get a few feedbacks from the project. In SRS, it shows there is not obvious relationship between each requirement, so each requirement can be implemented as one separated increment. In this way, it will have more testing phase, give clients more feedbacks.

2) **Communication between subteam:**

In SPMP, communication between subteams will be primarily by regular subteam leader/management meetings and then the subteam leaders pass relevant details to the rest of their subteam. Thus, there are not enough communications between different subteam members, which may have a bad impact on improving the system's performance quality. Team manager can provide multiple different platforms for more communication between the team members (e.g. online technique communication).

But the advantages of incremental model are still greater than the disadvantages, so in this essay it still regards choosing incremental model is a right one.

7. Reference

[1] Team Aura. (2011, April 7). Software Requirement Specification.

[2] Team Aura. (2011 May 17). Software Project Management Plan.